Simulation

R supports simulation from many distributions, including all the ones we have encountered. The general pattern of usage is that each distribution has a corresponding function that is called with the sample size an argument, and further arguments specifying parameters. The function returns the simulated observations as a vector.

Simulate Random variables with a given Distribution

Starting Point :- To simulate uniform $(0,1)$ random variables. [Assume we know how to do this]

$\boxed{\text{Random number generator}}$

- R – runif$(\cdot\cdot)$ = does it
- Random number Tables. [classical]
  – Given U – $U \sim$ uniform $(0,1)$

Example 1 :- $X \sim$ Bernoulli (p) $\quad 0 < p < 1$ – Simulate.

(i) Generate U

(ii) $$X = \begin{cases} 1 & \text{if } U < p \\ 0 & \text{if } U \geq p \end{cases}$$

[Check :- $\mathbb{P}(X=1) = \mathbb{P}(U < p) = p$

$U \sim$ uniform$(0,1)$

$\mathbb{P}(X=0) = \mathbb{P}(U \geq p) = 1-p$

Example 2 :- $X \sim$ Binomial (n, p) $\quad 0 < p < 1$ – Simulate.

Fact :- $X = \sum_{i=1}^{n} X_i \qquad X_i \sim$ Bernoulli (p)

(i) Generate $u_1, u_2, \ldots u_n$ $\quad u_i \sim$ iid Uniform $(0,1)$

$$X_i = \begin{cases} 1 & \text{if } u_i < p \\ 0 & \text{if } u_i \geq p \end{cases}$$

$$X = \sum_{i=1}^{n} X_i$$

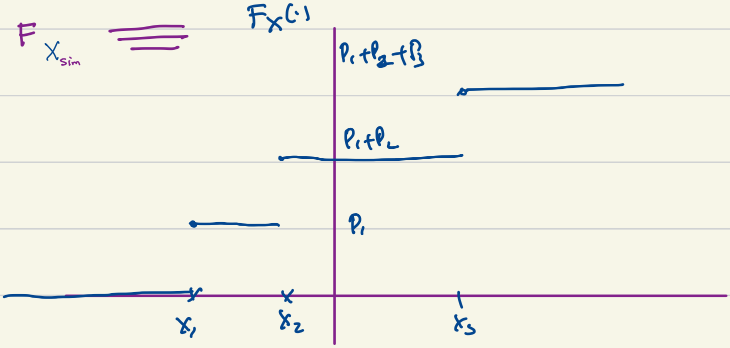
[Ex: $X \sim$ Binomial (n,p)]

Example 2 :- $\{p_i\}_{i=1}^{N}$; $\mathbb{P}(X=x_i) = p_i$; X – Simulate.
$x_1 < x_2 < \ldots < x_n$

(i) Generate u.

(ii)
$$\boxed{X_{sim} = \begin{cases} x_1 & u < p_1 \\ x_k & k \geq 2 \quad \sum_{i=1}^{k-1} p_i \leq u < \sum_{i=1}^{k} p_i \end{cases}}$$

$F_{X_{sim}} \equiv F_X(\cdot)$



$$\Rightarrow \quad X \overset{d}{\sim} X_{sim}$$

Binomial-simulation

30 Binomial$(100, 0.75)$ samples can be generated by

```
>   rbinom(30, size = 100, prob = 0.75)
 [1] 82 80 71 75 85 74 76 82 79 67 81 78 78 73 80 73 78 84
[26] 77 72 76 76 72
>   x <- rbinom(30, size = 100, prob = 0.75)
> mean(x)
[1] 74.93333
> sum(x >= 75) / length(x)
[1] 0.5
```

Example 3 :- X has p.d.f f - Simulate X

Assume: $F_X(\cdot) = \int_{-\infty}^{\cdot} f_X(x)\,dx$ is a strictly increasing function.

$\Rightarrow \quad F_X^{-1} : [0,1] \to \mathbb{R}$ exists.

(i) Generate U

(ii) Set $X_{sim} = F_X^{-1}(U)$

$$F_{X_{sim}}(x) = \mathbb{P}(X_{sim} \leq x) = \mathbb{P}(F_X^{-1}(U) \leq x)$$
$$= \mathbb{P}(U \leq F_X(x))$$
$$= F_X(x)$$

$$\Rightarrow \quad X_{sim} \stackrel{d}{=} X$$

Catch :- $X \sim N(0,1)$

$$F_X(x) = \int_{-\infty}^{x} \frac{e^{-\frac{y^2}{2}}}{\sqrt{2\pi}}\,dy$$

$F_X(\cdot)$ is not available explicitly !

For Normal distribution - Box — Mueller

(i) Generate $U_1, U_2 \overset{d}{=}$ Uniform $[0,1)$

(ii) $X = \sqrt{-2\ln(U_1)}\ \cos(2\pi U_2)$

$Y = \sqrt{-2\ln(U_1)}\ \sin(2\pi U_2)$

Check :- X, Y are iid Normal $(0,1)$

Example 3a :- X has Exponential (1) - Simulate X

(i) - Generate U

(ii) $F_X(x) = \int_0^x 1\, e^{-y}\, dy$

$= 1 - e^{-x}$

$F_X^{-1}(x) = -\ln(1-x)$

Set $X_{sim} = -\ln(1-U)$

Replicate

R has a useful function called `replicate` that allows us to repeat such an experiment several times.

```
>   replicate(15, {
+     x <- rbinom(30, size = 100, prob = 0.75)
+     mean(x)
+   })
> replicate(15, {
+   x <- rbinom(30, size = 100, prob = 0.75)
+   sum(x >= 75) / length(x)
+ })
```

We can simulate 50 observations from the Exp(1) distribution using the following R code.

```
>   -log(1 - runif(50))
 [1] 3.339399043 3.839420116 0.973062642 0.257627141 0.6732
 [7] 1.220089130 1.582697840 0.649338389 0.002144902 1.6232
[13] 1.447589888 0.590886612 3.448617803 0.351869674 1.3436
[19] 1.235740080 2.516965060 1.169089934 0.320640405 0.1163
[25] 0.183102471 0.070909704 0.001620309 0.320198287 0.0831
[31] 0.351011481 0.461761071 2.209017785 2.064828471 0.0997
[37] 0.501703163 2.795699029 0.138608841 1.080552074 0.6363
[43] 0.780775691 0.051455262 1.746644181 0.314377586 1.7221
[49] 0.516125096 0.228921892
```

==